HEWLETT PACKARD COMPANY
Intellectual Property Administration
P. O. Box 272400
Fort Collins, Colorado 80527-2400

PATENT APPLICATION

ATTORNEY DOCKET NO. 10010858-1

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Guangrui Fu et al.

Confirmation No.: 1851

Application No.: 10/057,124

Examiner: Aaron D. Matthew

Filing Date: 01-24-2002

Group Art Unit: 2114

Title: System and Method for Memory Failure Recovery Using Lockstep Processes

Mail Stop Appeal Brief-Patents
Commissioner For Patents
PO Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Sir:

Transmitted herewith is the Appeal Brief in this application with respect to the Notice of Appeal filed on May 17, 2005.

The fee for filing this Appeal Brief is (37 CFR 1.17(c)) $500.00.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

( ) (a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d) for the total number of months checked below:

( )	one month	$120.00
( )	two months	$450.00
( )	three months	$1020.00
( )	four months	$1590.00

( ) The extension fee has already been filled in this application.

(X) (b) Applicant believes that no extension of time is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account **08-2025** the sum of **$500.00**. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees. A duplicate copy of this sheet is enclosed.

(X) I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, Alexandria, VA 22313-1450. Date of Deposit: 07-14-2005

OR

( ) I hereby certify that this paper is being transmitted to the Patent and Trademark Office facsimile number _____ on _____

Number of pages:

Typed Name: **Ginger Yount**

Signature: Ginger Yount

Respectfully submitted,

Guangrui Fu et al.

By Dan C. Hu

Dan C. Hu

Attorney/Agent for Applicant(s)
Reg. No. **40,025**

Date: **07-14-2005**

Telephone No.: **(713) 468-8880**

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Guangrui Fu et al.	§	Group Art Unit:	2114
Serial No.:	10/057,124	§	Examiner:	Aaron D. Matthew
Filed:	January 24, 2002	§		
For:	System and Method for Memory Failure Recovery Using Lockstep Processes	§	Atty. Dkt. No.:	10010858-1 (HPC.0203US)

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF PURSUANT TO 37 C.F.R § 41.37

Sir:

The final rejection of claims 1-15 and 17-30 is hereby appealed.

I. REAL PARTY IN INTEREST

The real party in interest is the Hewlett-Packard Development Company.

II. RELATED APPEALS AND INTERFERENCES

None.

Date of Deposit: July 14, 2005

I hereby certify under 37 CFR 1.8(a) that this correspondence is being deposited with the United States Postal Service as **first class mail** with sufficient postage on the date indicated above and is addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313.

Ginger Yount

Ginger Yount

III. STATUS OF THE CLAIMS

Claims 1-15 and 17-30 have been finally rejected and are the subject of this appeal. Claim 16 has been cancelled.

IV. STATUS OF AMENDMENTS

No amendments were filed after the final rejections.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

The following provides a concise explanation of the subject matter defined in each of the independent claims involved in the appeal, referring to the specification by page and line number and to the drawings by reference characters, as required by 37 C.F.R. § 41.37(c)(1)(v). Each element of the claims is identified by a corresponding reference to the specification and drawings where applicable. Note that the citation to passages in the specification and drawings for each claim element does not imply that the limitations from the specification and drawings should be read into the corresponding claim element.

Independent claim 1 recites a method for memory failure recovery in a computer, comprising:

> maintaining a predetermined number of duplicate and primary processes in the computer (*see e.g.,* Specification, p. 8, lines 14-22; p. 9, lines 8-22; p. 14, lines 5-22; p. 15, lines 1-9; Fig. 2: 208, 210, 224; Fig. 6:502, 504, 506, 602, 610, 620, 604, 612, 614);
>
> keeping the processes in synchronization (*see e.g.,* Specification, p. 10, line 14-p. 11, line 21; p. 14, line 18-p. 15, line 14);
>
> managing the processes so that a single process image is presented to an external environment (*see e.g.,* Specification, p. 10, line 23-p. 11, line 5; p. 11, lines 22-24);

detecting a computer exception which affects one of the processes (*see e.g.,* Specification, p. 12, line 25-p. 13, line 7); and

terminating the affected process (*see e.g.,* Specification, p. 13, lines 8-12).

Independent claim 15 recites a method for memory failure recovery, comprising:

maintaining a predetermined number of duplicate and primary processes (*see e.g.,* Specification, p. 8, lines 14-22; p. 9, lines 8-22; p. 14, lines 5-22; p. 15, lines 1-9; Fig. 2: 208, 210, 224; Fig. 6:502, 504, 506, 602, 610, 620, 604, 612, 614);

keeping the processes in synchronization (*see e.g.,* Specification, p. 10, line 14-p. 11, line 21; p. 14, line 18-p. 15, line 14);

managing the processes so that a single process image is presented to an external environment (*see e.g.,* Specification, p. 10, line 23-p. 11, line 5; p. 11, lines 22-24);

detecting a computer system exception which affects one of the processes (*see e.g.,* Specification, p. 12, line 25-p. 13, line 7); and

terminating the affected process (*see e.g.,* Specification, p. 13, lines 8-12);

wherein the maintaining element includes,

identifying a primary process (*see e.g.,* Specification, p. 8, lines 14-22; p. 14, lines 4-9);

monitoring a fault-tolerance value corresponding to the primary process (*see e.g.,* Specification, p. 9, lines 1-8; p. 14, lines 4-12); and

setting a number of duplicate processes equal to the fault-tolerance value (*see e.g.,* Specification, p. 14, line 20-p. 15, line 9); and

wherein the managing element includes,

permitting only one of the processes to perform a system call to an external environment (*see e.g.,* Specification, p. 10, lines 23-26).

Independent claim 17 recites a computer-usable medium embodying computer program code for commanding a computer to perform memory failure recovery comprising:

maintaining a predetermined number of duplicate and primary processes in the computer (*see e.g.,* Specification, p. 8, lines 14-22; p. 9, lines 8-22; p. 14, lines 5-

22; p. 15, lines 1-9; Fig. 2: 208, 210, 224; Fig. 6:502, 504, 506, 602, 610, 620, 604, 612, 614);

keeping the processes in synchronization (*see e.g.,* Specification, p. 10, line 14-p. 11, line 21; p. 14, line 18-p. 15, line 14);

managing the processes so that a single process image is presented to an external environment (*see e.g.,* Specification, p. 10, line 23-p. 11, line 5; p. 11, lines 22-24);

detecting a computer system exception which affects one of the processes (*see e.g.,* Specification, p. 12, line 25-p. 13, line 7); and

terminating the affected process (*see e.g.,* Specification, p. 13, lines 8-12).

Independent claim 22 recites a system for memory failure recovery in a computer, comprising:

means for maintaining a predetermined number of duplicate and primary processes in the computer (*see e.g.,* Specification, p. 8, lines 14-22; p. 9, lines 8-22; p. 14, lines 5-22; p. 15, lines 1-9; Fig. 2: 208, 210, 224; Fig. 6:502, 504, 506, 602, 610, 620, 604, 612, 614);

means for keeping the processes in synchronization (*see e.g.,* Specification, p. 10, line 14-p. 11, line 21; p. 14, line 18-p. 15, line 14);

means for managing the processes so that a single process image is presented to an external environment (*see e.g.,* Specification, p. 10, line 23-p. 11, line 5; p. 11, lines 22-24);

means for detecting a computer exception which affects one of the processes (*see e.g.,* Specification, p. 12, line 25-p. 13, line 7); and

means for terminating the affected process (*see e.g.,* Specification, p. 13, lines 8-12).

Independent claim 23 recites a system for memory failure recovery, comprising:

a primary process memory space hosting a primary process (*see e.g.,* Specification, p. 9, line 24-p. 10, line 4);

a duplicate process memory space hosting a duplicate process corresponding to the primary process (*see e.g.,* Specification, p. 9, line 23-p. 10, line 4);

a synchronization buffer (*see e.g.*, Fig. 2: 214; Fig. 6: 608, 618, 624) for keeping the duplicate process in synchronization with the primary process (Specification, p. 10, line 14-p. 11, line 21; p. 14, line 18-p. 18, line 14);

a processor for generating an exception signal in response to detection of a memory failure condition which affects the primary process (*see e.g.*, Specification, p. 12, line 25-p. 13, line 7); and

an operating system for receiving the exception signal, terminating the affected primary process, and maintaining a predetermined number of primary and duplicate processes (*see e.g.*, Specification, p. 13, lines 8-23).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

A. **Claims 1, 3-5, 7, 8, 12, 17, and 19-22 Were Rejected Under 35 U.S.C. § 103 Over U.S. Patent No. 6,195,760 (Chung) and U.S. Patent No. 6,684,396 (Brittain).**

B. **Claims 2 and 18 Were Rejected Under 35 U.S.C. § 103 Over Chung in View of Brittain and U.S. Patent No. 6,684,346 (Tu).**

C. **Claims 23-25 Were Rejected Under 35 U.S.C. § 103 Over Chung in View of U.S. Patent No. 6,247,143 (Williams) and Tu.**

D. **Claims 11, 26, and 29 Were Rejected Under 35 U.S.C. § 103 Over Chung in View of Brittain and Williams.**

E. **Claims 9 and 10 Were Rejected Under 35 U.S.C. § 103 Over Chung in View of Brittain and U.S. Patent No. 6,263,452 (Jewett).**

F. **Claim 6 Was Rejected Under 35 U.S.C. § 103 Over Chung in View of Brittain and U.S. Patent Application Publication No. 2003/0061530 (Hayden).**

G. **Claim 15 Was Rejected Under 35 U.S.C. § 103 Over Chung in View of "Official Notice."**

H. **Claims 13 and 14 Were Rejected Under 35 U.S.C. § 103 Over Chung in View of Brittain and "Official Notice."**

I. **Claim 27 Was Rejected Under § 103 Over Chung in View of "Official Notice" and Williams.**

J. **Claim 28 Was Rejected Under § 103 Over Chung in View of "Official Notice" and Brittain.**

K. **Claim 30 Was Rejected Under § 103 Over Chung in View of Williams, Tu and Brittain.**

VII. ARGUMENT

A. Claims 1, 3-5, 7, 8, 12, 17, and 19-22 Were Rejected Under 35 U.S.C. § 103 Over U.S. Patent No. 6,195,760 (Chung) and U.S. Patent No. 6,684,396 (Brittain).

1. Claims 1, 4, 5, 7, 12, 17, and 20-22.

Independent claims 1, 17, and 22 were rejected as being obvious over the asserted combination of Chung and Brittain. Appellant respectfully submits that the Examiner has failed to establish a *prima facie* case of obviousness for at least the reason that no motivation or suggestion existed to combine the teachings of Chung and Brittain. *See* M.P.E.P. § 2143 (8th ed., Rev. 2), at 2100-129; *see also In re Fritch*, 972 F.2d 1260, 1264, 23 U.S.Q.P.2d 1780 (Fed. Cir. 1992) (an examiner can satisfy the burden of establishing a *prima facie* case of obviousness *only by showing some objective teaching* in the prior art or that knowledge generally available to one of ordinary skill in the art would lead one to combine the relevant teachings of the references).

Claim 1 recites maintaining a predetermined number of duplicate and primary processes in *a* computer, in combination with the remaining elements of claim 1. In contrast, Chung teaches the provision of primary and backup application modules on *multiple host computers* in the networked environment. As depicted in Fig. 1, and described in columns 3 and 4, of Chung, a primary copy of an application module runs on one host computer, while backup application modules reside "on at least one of the other host computers in an idle state awaiting later execution, or are running as a [sic] backup copies or second primary copies of application modules." Chung, 4:10-15. Chung's fault-tolerant mechanism is focused on replication for "distributed applications in a network" (*see* Title of Chung). As further stated by Chung, "[r]eplication of the application module on other host computers in the network is a well-known technique that can be used to improve reliability and availability of the application module." Chung, 1:33-35. The provision of application modules on multiple host computers enables

fail-over from the application module in one host computer to a backup application module in another host computer on the network. *See* Chung, 5:8-20 (describing two solutions of migrating from one host computer to another host computer in response to failure of an application module).

However, Chung does not disclose maintaining the predetermined number of duplicate and primary processes in *a* computer, as recited in claim 1. The Examiner has agreed with this point, conceding that Chung fails to teach that the predetermined number of duplicate and primary processes are in a single computer. 3/14/2005 Office Action at 3. However, the Examiner relied upon Brittain as teaching the claimed element that is not taught by Chung. The Examiner argued that a person of ordinary skill in the art would have been motivated to combine the teachings of Chung and Brittain "because Brittain shows a need in the art for providing fault-tolerance in single computer systems, by running redundant copies of software processes." *Id.* at 4.

There are several reasons why the Examiner's assessment of obviousness is not correct.

First, the Examiner's obviousness rejection appears to have been premised on an incorrect understanding of Brittain. The Examiner incorrectly states: "Though Brittain also teaches an embodiment wherein said duplicate processes run on separate computers, Brittain discloses an embodiment wherein all copies of a process reside on a single computer, (see Abstract, and col. 4, lines 61-64)." 3/14/2005 Office Action at 3. Contrary to the Examiner's understanding, Brittain does not teach an embodiment where duplicate processes run on separate computers; rather, Brittain only teaches a single system that has multiple CPUs, where a primary process and backup process can run on multiple respective CPUs or on one CPU, *in the same system.* The passage in column 4, lines 61-64, cited by the Examiner refers to running multiple

copies of a process on a single CPU rather than on multiple CPUs. Because Brittain teaches the provision of primary and backup copies of software processes that are run on one CPU or multiple CPUs in a *single* computer, Brittain provides no suggestion that its mechanism can be used to substitute the Chung arrangement that involves provision of primary and duplicate application modules on *separate host computers*.

A further reason that there existed no motivation or suggestion to combine Chung and Brittain is that there is clearly no desirability suggested in either Chung or Brittain of modifying Chung to provide the backup application modules of Chung into the same host computer as the primary copy of an application module. Note that Chung is related to failure recovery that contemplates switching to a completely different host computer in case of failure of an application module in one host computer. There is no desirability to modify Chung to incorporate primary and backup application modules into a single host computer, as that would be counter to the multi-host solution proposed by Chung.

It is well established law that "[t]he mere fact that the prior art could be so modified would not have made the modification **obvious** unless the prior art suggested the **desirability** of the modification." *In re Gordon,* 733 F.2d 900, 902, 221 U.S.P.Q. 1125 (Fed. Cir. 1984) (emphasis added). As the Federal Circuit has stated, "virtually all [inventions] are combinations of old elements." *In re Rouffet*, 149 F.3d 1350, 1357, 47 U.S.P.Q.2d 1453 (Fed. Cir. 1998). "Most, if not all, inventions are combinations and mostly of old elements." *Id.*

> Therefore an examiner may often find every element of a claimed invention in the prior art. If identification of each claimed element in the prior art were sufficient to negate patentability, very few patents would ever issue. Furthermore, rejecting patents solely by finding prior art corollaries for the claimed elements would permit an examiner to use the claimed invention itself as a blueprint for piecing together elements in the prior art to defeat the patentability of the claimed invention. Such an approach would be 'an illogical and inappropriate process by which to determine patentability.'

Id.

Here, Chung explicitly teaches the provision of primary and backup copies of application modules on *separate* host computers. Maintaining primary and backup copies of application modules as taught by Chung is a much more expensive solution than maintaining the primary and backup copies of application modules in a single computer. Yet, despite the more expensive cost of maintaining primary and backup copies of application modules on separate host computers, Chung nevertheless proposes a solution that employs such an arrangement, with no hint given anywhere in Chung of any desirability to employ the primary and backup application modules in a single computer.

Brittain does not provide any suggestion of a modification of Chung to achieve the claimed invention. In fact, Brittain relates to upgrading software to a different version in a computer running multiple copies of the software. Brittain teaches performing a software upgrade in a system having a primary copy and a backup copy, where the software upgrade is performed by running a third copy of the software. Brittain, 3:29-36. A version manager in the system of Brittain then forces a switch in the primary copy from the original primary copy to the third copy. Brittain, 3:62-67. Brittain thus teaches a purportedly improved technique of performing upgrades in a computer having multiple copies of a software—however, there is no hint or suggestion provided anywhere in Brittain of modifying the multi-host computer solution taught by Chung to achieve the claimed invention.

Without the requisite motivation or suggestion to combine the teachings of Chung and Brittain to achieve the claimed invention, the Examiner has engaged in impermissible hindsight in using the claimed invention as an instruction manual or template to piece together teachings of prior art references. *See In re Fritch*, 972 F.2d at 1266 (one cannot use impermissible hindsight to pick and choose among isolated disclosures in the prior art to achieve the claimed invention).

In view of the foregoing, it is respectfully submitted that a *prima facie* case of obviousness has not been established with respect to independent claim 1. A *prima facie* case of obviousness has not been established with respect to independent claims 17 and 22 for similar reasons.

Reversal of the final rejection of the above claims is respectfully requested.

2. Claims 3 and 19.

Claims 3 and 19 depend from independent claims 1 and 17, respectively, and thus are allowable for at least the same reasons as the corresponding independent claims.

Moreover, claim 3 recites allocating a new memory space in memory hardware in the computer to each of the duplicate processes, which is separate from a memory space in the memory hardware allocated to the primary process. Clearly, there existed absolutely no need in Chung to allocate separate memory spaces in the same memory hardware in a computer to the backup and primary application modules of Chung. That is because Chung uses completely separate host computers – therefore, a person of ordinary skill in the art would have recognized that allocating separate memory spaces in the same memory hardware in a computer for the multiple application modules would have been completely unnecessary.

The following constitutes the Examiner's rejection of claim 3: "see Brittain, col. 6, lines 1-4." 3/14/2005 Office Action at 5. No explanation was provided regarding why a person of ordinary skill in the art would have been motivated to allocate separate memory spaces in memory hardware in a computer to duplicate and primary processes in light of the fact that Chung already teaches separate host computers running separate copies of the application modules. *Id.* at 5.

A *prima facie* case of obviousness has not been established with respect to claims 3 and 19 for at least the above additional reasons. Reversal of the final rejection of the above claims is respectfully requested.

3. Claim 8.

Claim 8 depends indirectly from claim 1, and is thus allowable for at least the same reasons as claim 1. Moreover, claim 8 recites deleting a duplicate process if a number of duplicate processes is more than the fault-tolerance value. In the rejection of claim 8, the Examiner stated that "[i]t is inherent that, if the total number of duplicate processes is to be maintained at an amount equal to the degree of replication, any additional duplicate processes beyond that amount must be deleted." 3/14/2005 Office Action at 6.

The Examiner has failed to satisfy the requirements set forth in the M.P.E.P. for a rejection based on inherency. *See* M.P.E.P. § 2112. As stated by the M.P.E.P., "[t]he fact that a certain result or characteristic may occur or be present in the prior art is not sufficient to establish the inherency of that result or characteristic." *Id.* at 2100-54. "To establish inherency, the extrinsic evidence 'must make clear that the missing descriptive matter is *necessarily* present in the thing described in the reference, and that it would be so recognized by persons of ordinary skill.'" *Id.* "In relying upon the theory of inherency, the examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art." *Id.* at 2100-55. Here, the Examiner has failed to establish that deletion of a duplicate process if the number of duplicate processes is more than a fault-tolerance value is *necessarily* present in Chung. Chung teaches that a registration request can specify a degree of replication for copies of application modules on respective host computers. Chung, 4:53-65. There is no indication in Chung that it

is even possible to start a number of copies of an application module that is greater than the degree of replication specified in the registration request. In fact, the teachings of Chung seem to indicate that the number of copies of the application module that are started is equal to this degree of replication, with no suggestion provided anywhere in Chung that a larger number of backup copies can be started. Thus, the element recited in claim 8 regarding the deletion of a duplicate process if the number of duplicate processes is more than the fault-tolerance value clearly is not necessarily present in Chung. Therefore, the *prima facie* case of obviousness with respect to claim 8 is defective for this additional reason.

Reversal of the final rejection of claim 8 is respectfully requested.

B. Claims 2 and 18 Were Rejected Under 35 U.S.C. § 103 Over Chung in View of Brittain and U.S. Patent No. 6,684,346 (Tu).

1. Claims 2 and 18.

Claims 2 and 18, which depend from independent claims 1 and 17, respectively, were rejected as being obvious over Chung, Brittain, and Tu. In view of the fact that the obviousness rejection of base claims 1 and 17 over Chung and Brittain is defective, it is respectfully submitted that the obviousness rejection of claims 2 and 18 over Chung, Brittain, and Tu is also defective.

Moreover, Tu provides no suggestion whatsoever of detecting a memory failure that affects one of the duplicate and primary processes, and terminating the affected process (affected by the memory failure). Tu notes that a parity error, such as a cache memory being stuck at some value, is handled by a machine check abort handler. Tu, 5:9-14. Tu teaches that in a multiprocessor environment, MCA handling for the parity error may require synchronization of processors and arbitration for shared resources. Tu, 5:25-27. Thus, what Tu would have

suggested to a person of ordinary skill in the art would be that a memory failure would require synchronization of processors and arbitration for shared resources. Tu does not suggest that an affected process be terminated in response to detecting the memory failure. Therefore, no motivation or suggestion existed to combine the teachings of Chung, Brittain, and Tu to achieve the claimed invention. A *prima facie* case of obviousness with respect to claims 2 and 18 has therefore not been established for this additional reason.

Reversal of the final rejection of the above claims is respectfully requested.

C. Claims 23-25 Were Rejected Under 35 U.S.C. § 103 Over Chung in View of U.S. Patent No. 6,247,143 (Williams) and Tu.

1. Claims 23-25.

Independent claim 23, and its dependent claims 24 and 25, were rejected as being obvious over Chung, Williams and Tu. It is respectfully submitted that a *prima facie* case of obviousness has not been established with respect to claim 23 for at least two reasons: (1) there existed no motivation or suggestion to combine the reference teachings; and (2) even if the teachings of Chung, Williams, and Tu are combined, the hypothetical combination of such references does not teach or suggest *all* of the claimed elements. *See* M.P.E.P. § 2143, at 2100-129.

Point (2) is addressed first. Note that claim 23 recites an operating system for receiving an exception signal, *terminating the affected primary process*, and maintaining a predetermined number of primary and duplicate processes. The Examiner stated that "a step of terminating a process is inherently included in a step of restarting a process." 3/14/2005 Office Action at 9-10, 28 (citing column 2, lines 39-41 and 45-49, and column 3, lines 24-31, of Chung). Appellant respectfully disagrees with this statement. The watchdog daemon described in Chung is able to

detect that a host computer has crashed or has hung. However, there is absolutely no mention whatsoever that the watchdog daemon performs the termination of an application module. *See* Chung, 7:35-63. In Chung, it is assumed that the primary application module has already crashed – therefore, there is no need for the watchdog daemon to terminate the affected application module. Furthermore, attempting to restart an application module is not the same as terminating the application module. In fact, restarting an application module is the opposite of terminating the application module. The obviousness rejection is defective for at least this reason.

Additionally, there simply did not exist any motivation or suggestion to combine at least the teachings of Chung and Williams. As discussed above, Chung relates to a networked environment in which backup copies of an application module are kept on separate host computers. Fail-over can occur from one host computer to another host computer in response to detection of failure of one of the host computers. The Examiner conceded that Chung fails to teach a synchronization buffer for keeping the duplicate process in synchronization with this primary process. 3/14/2005 Office Action at 10. However, the Examiner relied upon Williams as teaching this feature. It is respectfully submitted that there existed no motivation or suggestion to incorporate the buffers described in the multiprocessor computer system of Williams into Chung's networked environment.

There is no need to employ the buffering mechanism in a multiprocessor system described in Williams for synchronizing application modules running on multiple host computers described in Chung. The buffering mechanism described in Williams enables interaction among multiple processors of a multiprocessing system of Williams. However, it is clear that a person of ordinary skill in the art would not have been motivated to employ the buffering mechanism

described in Williams in a network environment as taught by Chung. Such a buffering mechanism is simply not applicable to a network. The buffering mechanism described in Williams is in the context of a single computer system that has multiple processors – no indication whatsoever is provided in Williams that its technique can be applied to a network environment having multiple host computers connected to a network. In view of the foregoing, it is respectfully submitted that there existed no motivation to combine the teachings of Chung, Williams, and Tu.

The Examiner also conceded that Chung fails to teach a memory failure condition. The Examiner relied upon Tu as teaching this element. 3/14/2005 Office Action at 10. As discussed above, although Tu refers to handling memory failures, Tu describes this handling as involving synchronization of multiple processors. Tu, 5:25-30. Tu provides no suggestion whatsoever of a processor for generating an exception signal in response to detecting a memory failure condition that affects the primary process, and an operating system for receiving the exception signal, and terminating the affected primary process. This deficiency of Tu is another basis that no motivation or suggestion existed to combine the teachings of Chung, Williams, and Tu.

In view of the foregoing, it is respectfully submitted that a *prima facie* case of obviousness has not been established with respect to claim 23 and its dependent claims. Reversal of the final rejection of the above claims is respectfully requested.

D. Claims 11, 26, and 29 Were Rejected Under 35 U.S.C. § 103 Over Chung in View of Brittain and Williams.

1. Claims 11, 26, and 29.

Dependent claims 11, 26 (which depend from claim 1), and 29 (which depends from independent claim 17) were rejected as being obvious over Chung, Brittain, and Williams. In

view of the fact that a *prima facie* case of obviousness has not been established with respect to claims 1 and 17 over Chung and Brittain, the obviousness rejections of claims 11, 26, and 29 is defective.

Moreover, as discussed above, Williams teaches a synchronization mechanism that is applicable to a single computer system that has multiple processors. There is no teaching or suggestion anywhere within Williams, Chung, or Brittain of using the synchronization mechanism of Williams in the network environment having multiple host computers in Chung. Therefore, no motivation or suggestion existed to combine the teachings of Chung, Brittain, and Williams to achieve the invention of claims 11, 26, and 29.

For the foregoing reasons, reversal of the final rejections of the above claims is respectfully requested.

E. Claims 9 and 10 Were Rejected Under 35 U.S.C. § 103 Over Chung in View of Brittain and U.S. Patent No. 6,263,452 (Jewett).

1. Claims 9 and 10.

Claims 9 and 10, which depend from claim 1, were rejected as being obvious over Chung, Brittain, and Jewett. In view of the fact that a *prima facie* case of obviousness has not been established with respect to claim 1 over Chung and Brittain, the obviousness rejection of claims 9 and 10 over Chung, Brittain, and Jewett is also defective.

Moreover, with respect to claims 9 and 10, the Examiner conceded that the teachings of Chung and Brittain fail to disclose processes that are synchronized upon receipt of data or signals from an external environment. Reliance was made by the Examiner on Jewett as teaching this element. 3/14/2005 Office Action at 17. Note that, like Williams, Jewett teaches CPUs within *a single* computer system based on detecting events such as memory references. Jewett, 2:36-56.

There is absolutely no indication or suggestion anywhere in Jewett, or in Chung or Brittain, of applying the synchronization technique that relates to multiple CPUs in a single multiprocessor system, as taught by Jewett, to a network environment having multiple host computers, as taught by Chung. Therefore, no motivation or suggestion existed to combine the teachings of Chung, Brittain, and Jewett.

Reversal of the final rejection of the above claims is respectfully requested.

F. Claim 6 Was Rejected Under 35 U.S.C. § 103 Over Chung in View of Brittain and U.S. Patent Application Publication No. 2003/0061530 (Hayden).

1. Claim 6.

Claim 6, which depends indirectly from claim 1, was rejected as being obvious over Chung, Brittain, and Hayden. In view of the fact that a *prima facie* case of obviousness of claim 1 has not been established with respect to Chung and Brittain, the obviousness rejection of claim 6 over Chung, Brittain, and Hayden is also defective.

STATEMENT OF COMMON OWNERSHIP

Also, Hayden is disqualified as prior art under 35 U.S.C. § 103(c). The present application and Hayden were, at the time the invention of the present application was made, owned by, or subject to an obligation of assignment to, the same person.

Reversal of the final rejection of claim 6 is therefore respectfully requested.

G. Claim 15 Was Rejected Under 35 U.S.C. § 103 Over Chung in View of "Official Notice."

1. Claim 15.

Claim 15 was rejected as being obvious over Chung in view of "official notice." The Examiner conceded that Chung does not disclose that the managing element includes permitting

only one of the processes to perform a system call to an external environment. 3/14/2005 Office Action at 21. However, the Examiner took "official notice" that such a system call would have been well known in the art. *Id.*

In response to Appellant's challenge of the official notice and Appellant's request that the Examiner produce a reference to support the taking of the "official notice," the Examiner cited Swanberg. Note that claim 15 recites "permitting only one of the processes to perform a system call to an external environment." On the other hand, Swanberg relates to a system and method for using a 32-bit address for branching into a 64-bit target address. Swanberg, 1:9-11. Specifically, Swanberg describes executing 32-bit programs in a 64-bit environment and enabling 32-bit addresses to access regions within the 64-bit environment. Swanberg, 5:33-35. The 32-bit program includes dynamic links to runtime libraries that are referred to as "out of module" calls. Swanberg, 5:39-41. The library call or system call referred to in Swanberg does not relate to a system call to an external environment for the purpose of managing a predetermined number of duplicate and primary processes, as recited in claim 15. The library or system calls referred to in Swanberg are calls to runtime libraries from a 32-bit program in a 64-bit environment. Therefore, the reliance upon Swanberg as supporting the "official notice" taken by the Examiner is clearly improper, as Swanberg does not provide the required suggestion to modify Chung to achieve the claimed invention.

A *prima facie* case of obviousness has therefore not been established with respect to claim 15. Reversal of the final rejection of claim 15 is therefore respectfully requested.

H. Claims 13 and 14 Were Rejected Under 35 U.S.C. § 103 Over Chung in View of Brittain and "Official Notice."

1. Claims 13 and 14.

Claims 13 and 14, which depend from claim 1, were rejected as being obvious over Chung, Brittain, and "official notice." In view of the fact that a *prima facie* case of obviousness of claim 1 over Chung and Brittain has not been established, the obviousness rejections of claims 13 and 14 over Chung, Brittain, and "official notice" is defective.

Also, as discussed above with respect to claim 15, Swanberg, cited by the Examiner to support the taking of official notice, does not provide the required suggestion to modify Chung to achieve the claimed invention. Reversal of the final rejection of the above claims is therefore respectfully requested.

I. Claim 27 Was Rejected Under § 103 Over Chung in View of "Official Notice" and Williams.

1. Claim 27.

Claim 27, which depends from independent claim 15, was rejected as being obvious over Chung in view of "official notice" and Williams. In view of the fact that a *prima facie* case of obviousness has not been established with respect to base claim 15 over Chung and "official notice," the obviousness rejection of claim 27 is defective.

Moreover, as discussed above, there did not exist any motivation or suggestion to incorporate the teachings of Williams (relating to synchronization in a single computer system) into the network environment of Chung (which includes multiple host computers connected to a network). Therefore, no motivation or suggestion existed to combine the teachings of Chung, "official notice," and Williams to achieve the claimed invention.

Reversal of the final rejection of claim 27 is respectfully requested.

J. Claim 28 Was Rejected Under § 103 Over Chung in View of "Official Notice" and Brittain.

1. Claim 28.

Claim 28, which depends from claim 15, was rejected as being obvious over Chung, "official notice," and Brittain. In view of the fact that a *prima facie* case of obviousness has not been established with respect to claim 15 over Chung and "official notice," the obviousness rejection of claim 28 over Chung, "official notice," and Brittain is defective.

Also, as discussed above, no motivation or suggestion existed to combine the teachings of Chung and Brittain. Therefore, no motivation or suggestion existed to combine the teachings of Chung, "official notice," and Brittain. Reversal of the final rejection of claim 28 is therefore respectfully requested.

K. Claim 30 Was Rejected Under § 103 Over Chung in View of Williams, Tu and Brittain.

1. Claim 30.

Claim 30, which depends from claim 23, was rejected as being obvious over Chung, Williams, Tu, and Brittain. In view of the fact that a *prima facie* case of obviousness of independent claim 23 has not been established over Chung, Williams, and Tu, the obviousness rejection of claim 30 over Chung, Williams, Tu, and Brittain is defective.

Moreover, as discussed above, no motivation or suggestion existed to combine the teachings of Chung with Brittain. This is a further basis that no motivation or suggestion existed to combine the teachings of Chung, Williams, Tu, and Brittain.

Reversal of the final rejection of claim 30 is therefore respectfully requested.

VIII. CONCLUSION

In view of the foregoing, reversal of all final rejections and allowance of all pending claims is respectfully requested.

Respectfully submitted,

Date: July 14, 2005

Dan C. Hu
Registration No. 40,025
TROP, PRUNER & HU, P.C.
8554 Katy Freeway, Suite 100
Houston, TX 77024
Telephone: (713) 468-8880
Facsimile: (713) 468-8883

APPENDIX OF CLAIMS

The claims on appeal are:

1. A method for memory failure recovery in a computer, comprising:
maintaining a predetermined number of duplicate and primary processes in the computer;
keeping the processes in synchronization;
managing the processes so that a single process image is presented to an external environment;
detecting a computer exception which affects one of the processes; and
terminating the affected process.

2. The method of claim 1 wherein the detecting element includes detecting a memory failure.

3. The method of claim 1 further comprising:
allocating a new memory space in memory hardware in the computer to each of the duplicate processes, which is separate from a memory space in the memory hardware allocated to the primary process.

4. The method of claim 1 wherein the maintaining element includes:
identifying a primary process;
monitoring a fault-tolerance value corresponding to the primary process; and
setting a number of duplicate processes equal to the fault-tolerance value.

5. The method of claim 4 further comprising assigning a predetermined fault-tolerance value to a primary process.

6. The method of claim 4 further comprising dynamically modifying the fault-tolerance value of the primary process, in response to a computer command.

7. The method of claim 4 wherein the setting element includes adding one or more new duplicate processes, if the number of duplicate processes is less than the fault-tolerance value.

8. The method of claim 4 wherein the setting element includes deleting a duplicate process, if the number of duplicate processes is more than the fault-tolerance value.

9. The method of claim 1 wherein the keeping element includes synchronizing the processes upon receipt of data from an external environment.

10. The method of claim 1 wherein the keeping element includes synchronizing the processes upon receipt of signals from an external environment.

11. The method of claim 1 wherein the keeping element includes synchronizing the processes upon transmission by one of the processes to an external environment.

12. The method of claim 1 wherein the managing element includes permitting only one of the processes to transmit to an external environment.

13. The method of claim 1 wherein the managing element includes permitting only one of the processes to perform a system call to an external environment.

14. The method of claim 1 wherein the managing element includes permitting only one of the processes to perform a library call to an external environment.

15. A method for memory failure recovery, comprising:

maintaining a predetermined number of duplicate and primary processes;

keeping the processes in synchronization;

managing the processes so that a single process image is presented to an external environment;

detecting a computer system exception which affects one of the processes; and

terminating the affected process;

wherein the maintaining element includes,

identifying a primary process;

monitoring a fault-tolerance value corresponding to the primary process; and

setting a number of duplicate processes equal to the fault-tolerance value; and

wherein the managing element includes,

permitting only one of the processes to perform a system call to an external environment.

17. A computer-usable medium embodying computer program code for commanding a computer to perform memory failure recovery comprising:

maintaining a predetermined number of duplicate and primary processes in the computer;

keeping the processes in synchronization;

managing the processes so that a single process image is presented to an external environment;

detecting a computer system exception which affects one of the processes; and

terminating the affected process.

18. The medium of claim 17 wherein the detecting element includes detecting a memory failure.

19. The medium of claim 17 further comprising:

allocating a new memory space in memory hardware in the computer to each of the duplicate processes, which is separate from a memory space in the memory hardware allocated to the primary process.

20. The medium of claim 17 wherein the maintaining element includes:
identifying a primary process;
monitoring a fault-tolerance value corresponding to the primary process; and
setting a number of duplicate processes equal to the fault-tolerance value.

21. The medium of claim 17 wherein the managing element includes permitting only one of the processes to transmit to an external environment.

22. A system for memory failure recovery in a computer, comprising:
means for maintaining a predetermined number of duplicate and primary processes in the computer;
means for keeping the processes in synchronization;
means for managing the processes so that a single process image is presented to an external environment;
means for detecting a computer exception which affects one of the processes; and
means for terminating the affected process.

23. A system for memory failure recovery, comprising:
a primary process memory space hosting a primary process;
a duplicate process memory space hosting a duplicate process corresponding to the primary process;
a synchronization buffer for keeping the duplicate process in synchronization with the primary process;
a processor for generating an exception signal in response to detection of a memory failure condition which affects the primary process; and
an operating system for receiving the exception signal, terminating the affected primary process, and maintaining a predetermined number of primary and duplicate processes.

24. The system of claim 23, further comprising:

a buffer controller for permitting the processes to receive communications from an external environment while permitting only one of the processes to transmit to the external environment.

25. The system of claim 23, wherein the exception signal is a machine check abort signal.

26. The method of claim 1, further comprising:

the processes communicating, through a synchronization buffer, with an external environment,

wherein keeping the processes in synchronization is based on interaction between the processes and the external environment through the synchronization buffer.

27. The method of claim 15, wherein keeping the processes in synchronization is based on data or signals received from the external environment, the external environment including computer functionality outside the processes.

28. The method of claim 15, wherein the predetermined number of duplicate and primary processes are maintained in a computer.

29. The medium of claim 17, wherein the computer program code is for commanding the computer to further perform:

enabling the processes to communicate, through a synchronization buffer, with an external environment,

wherein keeping the processes in synchronization is based on interaction between the processes and the external environment through the synchronization buffer.

30. The system of claim 23, wherein the primary process, duplicate process, synchronization buffer, processor, and operating system are part of a computer.